

Essay

Vergleich des SIFT- und SURF-Algorithmus

Hausaufgabe der Vorlesung Computer Vision

Philipp Elsner, B.Sc.

18. Mai 2015

Abstract

In diesem Essay werden der „Scale Invariant Feature Transform“ (SIFT) Algorithmus und der „Speeded-Up Robust Feature“ (SURF) Algorithmus zur Merkmalerkennung und Beschreibung mit einander auf Genauigkeit und Robustheit, sowie Geschwindigkeit und Komplexität verglichen.

Dafür wird in den einzelnen Schritten der Algorithmen analysiert und verglichen. Es stellt sich heraus, dass der SURF-Algorithmus schneller arbeitet und robuster ist gegenüber rauschen als der SIFT-Algorithmus. Jedoch kann der SIFT Algorithmus genauere Ergebnisse liefert, insofern die Zeit kein kritischer Faktor ist.

1 Einleitung

Ein wichtiger Bereich der Computer Vision ist die Erkennung von markanten und gleichbleibenden Merkmalen in Bildern. Die Erkennung der Merkmale soll sowohl skalierungsinvariant, als auch rotationsinvariant sein. Gleichzeitig dürfen Einflüsse wie Rauschen, Belichtung, Bildverzerrung und Bildtransformationen die Merkmalerkennung und den Merkmalsabgleich nicht negativ beeinflussen. Diese Merkmale werden in einem Vektor abgespeichert. In einem Letzten Schritt können dann verschiedene Merkmalsvektoren abgeglichen werden zum Beispiel Vektoren von Bildern aus unterschiedlichen Perspektiven.

Der SIFT-Algorithmus (Scale invariant feature detector) [1,2] und Surf (Speed-Up Robust Features) [3,4] sind zwei Algorithmen die dafür entwickelt wurden Features aus Bildern zu extrahieren. Sie sollen in zwei Kernbereichen verglichen werden: Geschwindigkeit/ Komplexität und Genauigkeit/ Robustheit.

Der Vergleich soll zeigen, dass der SURF-Algorithmus schneller arbeitet, als der SIFT-Algorithmus [3]. Die Performance des SURF-Algorithmus, gegenüber dem SIFT-

Algorithmus jedoch nicht viel schlechter ist. Außerdem soll der SURF-Algorithmus gegenüber Rauschen unempfindlicher sein. Der SIFT-Algorithmus soll jedoch bei nicht zeitkritischen Anwendungen genauer sein. Diese Aussagen werde ich im Folgenden belegen.

2 Hauptteil

Bei beiden Algorithmen werden ungefähr die gleichen Grundschriffe ausgeführt, um Features zu extrahieren. Aber die Durchführung der einzelnen Schritte unterscheidet sich deutlich im Detail. Die Schlüsselmerkmalen werden folgendermaßen bestimmt:

1. Finden der Position von Extrema im Skalenraum
2. Erkennen der Schlüsselmerkmale
3. Zuordnen der Orientierung
4. Erstellen eines Deskriptors
5. Matching

Um beide Algorithmen zu vergleichen, werde ich alle Schritte kurz betrachten. Der Surf baut auf dem Sift auf. Er ist jedoch in den einzelnen Schritten auf Geschwindigkeit optimiert.

Bei beiden Algorithmen werden zunächst die Positionen von möglichen Schlüsselmerkmalen in einem Bild bestimmt. Lowe [2] verwendet zur Suche nach Merkmalspunkten die Differences of Gaussians (DoG) bzw. auch Laplacian of Gaussian (LoG), um den Scale-Space und die Extrema zu ermitteln. Um diesen Vorgang zu beschleunigen, setzt der Surf auf eine Methode durch Berechnung der Hessematrix. Durch den Einsatz von Boxfiltern [3, S. 3] und Integralbildern [3, S.3, Fig.1] wird die Berechnung deutlich beschleunigt. Diese Methode wurde schon von Viola und Jones [7] verwendet. Mit den Boxfiltern wird die Approximation der Hessematrix maximiert. Die Performance des Algorithmus ist gleich bis besser. Detektoren, die mit Hilfe der Hessematrix implementiert werden, weisen Schwächen im Bereich der Reproduzierbarkeit auf. Diese Schwachstelle tritt bei Rotationen um $\frac{\pi}{4}$ auf [3, S.4 Fig.3]. An dieser Stelle wird beim Surf ein Qualitätsverlust zu Gunsten von gesteigerter Geschwindigkeit hingenommen. Die Faltung wird durch die Diskretisierung und Beschneidung des Filters beschleunigt. Beide Algorithmen suchen Merkmale im Skalenraum jedoch wird beim Surf nicht das Bild für jede Oktave skaliert sondern der Boxfilter [3, S. 4 Fig.4]. Dadurch, dass bei jeder Oktave die Filtergröße und auch die Samplingintervalle verdoppelt werden, wird die Geschwindigkeit der Berechnung im Skalenraum deutlich verbessert. Da das Bild nicht skaliert wird sind die einzelnen Berechnungsschritte voneinander unabhängig und können laut Bay et. al [3] parallel zueinander berechnet werden. Dieser Schritt, um den Algorithmus zu beschleunigen, hat auch Nachteile die wiederum die Qualität beeinträchtigen. Die Boxfilter erhalten hohe Frequenzen die normalerweise bei niedriger skalierten Bildern herausgefiltert worden wären. Dadurch wird die Skalierungsinvarianz eingeschränkt [3, S. 4]. Um die Schlüsselmerkmale zu extrahieren wird beim Surf-Algorithmus in einer 3x3x3

Nachbarschaft ein schnelles Verfahren von Neubeck und Van Gool verwendet[6]. Beim SIFT-Algorithmus wird ein Pixel mit 26 Nachbarn verglichen [2, S.7, Fig. 2]. Ein weiterer Punkt in dem sich die Algorithmen unterscheiden sind die Deskriptoren. Der Merkmalsvektor hat 64 Dimensionen dadurch wird die Berechnungsgeschwindigkeit und die Robustheit deutlich gesteigert [3, S.5], während Lowe 128 Dimensionen verwendet. Dadurch bleibt der SIFT-Algorithmus genauer, da er mit einem doppelt so großen Merkmalsvektor arbeitet und der Surf-Algorithmus ist somit wieder schneller und robuster in Bezug auf Rauschen. Um eine Rotationsinvarianz zu erreichen wird die Orientierung der Merkmale bestimmt. Beim SIFT wird ein Gradientenhistogramm mit 36 Einträgen (360°) berechnet. Der Surf-Algorithmus berechnet das Gradientenhistogramm mit Hilfe der Impulsantworten der Haar Wavelets. Dieser Berechnungsschritt ist wieder aufgrund der Integralbilder schneller. Nach der Berechnung der Impulsantworten werden diese mit einer Gaußfunktion gewichtet. Dadurch wird der Algorithmus robuster gegenüber geometrischen Verformungen [3,S.7]. Um Informationen über die Polarität der Intensitätsänderungen mit einzubringen, wird auch die Summe der Absolutwerte für den Merkmalsvektor verwendet. Aufgrund dieser Tatsache, wird der Algorithmus robuster gegenüber Rauschen [3, S.8 Fig. 14]. Die Wavelets sind außerdem Helligkeitsinvariant [3, S.8].

3 Zusammenfassung

Wie schon von Bay et al [3] beschrieben, ist der SURF-Algorithmus wesentlich schneller, als der SIFT-Algorithmus. Der SIFT-Algorithmus ist jedoch robuster in Bezug auf Lokalisierungsfehler. Beide Algorithmen haben eine vergleichbare Genauigkeit. P. M. Panchal et al [5], haben die Algorithmen verglichen. Sie sind zu dem Schluss gekommen, dass der SURF-Algorithmus am schnellsten ist und der SIFT-Algorithmus in den meisten Fällen am robustesten ist. Sie kommen zu dem Schluss, dass es von der Anwendung abhängt, welchen Algorithmus man verwendet. Diese Tatsachen unterstreichen meine zu Beginn aufgestellte These.

Eine Anwendung des SIFT-Algorithmus sollte wie aus den oben genannten Argumenten ersichtlich ist, nicht zeitkritisch sein und eher auf eine hohe Robustheit und Genauigkeit abzielen. Zu diesen Anwendungen gehören die Bildregistrierung und das „Image Stitching“. Bei vielen Digitalen Spiegelreflexkameras und Smartphones gibt es solche Funktionen. Diese setzen aus mehreren Bildern ein Panoramabild zusammen. Da es bei dieser Anwendung zu einer Rotation und Skalierung der Bilder kommen kann ist der SIFT die beste Wahl.

Für den Surf-Algorithmus kann man sich vor allem zeitkritische Anwendungen wie maschinelles Sehen vorstellen. Fahrassistenzsysteme könnten ein mögliches Beispiel sein. Sie müssen in kurzer Zeit Objekte, Menschen oder andere Hindernisse verarbeiten können.

Literatur

- [1] D.G. Lowe. Object recognition from local scale-invariant features. In Computer vision, 1999. The proceedings of the seventh IEEE international conference on, Band 2, S.1150–1157. Ieee, 1999.
- [2] D.G. Lowe. Distinctive image features from scale-invariant keypoints. In International journal of computer vision, 60(2), S. 91–110, 2004.
- [3] Bay, Herbert, et al. "Speeded-up robust features (SURF)." *Computer vision and image understanding* 110.3 (2008): 346-359.
- [4] H. Bay, T. Tuytelaars und L. Van Gool. Surf: Speeded up robust features. In Computer vision–ECCV 2006, S. 404–417. Springer, 2006.
- [5] P. M. Panchal, S. R. Panchal, S. K. Shah, "[A Comparison of SIFT and SURF](#) ", International Journal of Innovative Research in Computer and Communication Engineering Vol. 1, Issue 2, April 2013
- [6] A. Neubeck und L. Van Gool. Efficient non-maximum suppression. In Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, Band 3, S. 850–855. IEEE, 2006.
- [7] P. Viola und M. Jones. Rapid object detection using a boosted cascade of simple features. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, Band 1, S. I–511. IEEE, 2001.